

NASA TM-88248, pt. I

NASA Technical Memorandum 88248

NASA-TM-88248-PT-1
19860020293

Transonic Navier-Stokes Wing Solution Using a Zonal Approach: Part I. Solution Methodology and Code Validation

J. Flores, T.L. Holst, Unver Kaynak, K. Gundy, and
S.D. Thomas

April 1986

LIBRARY COPY

APR 30 1986

LANGLEY RESEARCH CENTER
LIBRARY, NASA
HAMPTON, VIRGINIA



NF00953

NASA

National Aeronautics and
Space Administration

FOR REFERENCE

NOT TO BE TAKEN FROM THIS ROOM

Transonic Navier-Stokes Wing Solution Using a Zonal Approach: Part I. Solution Methodology and Code Validation

J. Flores,
T. L. Holst, Ames Research Center, Moffett Field, California
Unver Kaynak, Sterling Software, Palo Alto, California
K. Gundy, Ames Research Center, Moffett Field, California
S. D. Thomas, Sterling Software, Palo Alto, California

April 1986



National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

N86-29765 #

TRANSONIC NAVIER-STOKES WING SOLUTION USING A ZONAL APPROACH: PART 1. SOLUTION METHODOLOGY AND CODE VALIDATION

J. Flores*, T. L. Holst**

NASA Ames Research Center, Moffett Field, CA 94035

Unver Kaynak*

Sterling Software, Palo Alto, CA 94303

K. Gundy*

NASA Ames Research Center, Moffett Field, CA 94035

S. D. Thomas*

Sterling Software, Palo Alto, CA 94303

SUMMARY

A fast diagonalized Beam-Warming algorithm is coupled with a zonal approach to solve the three-dimensional Euler/Navier-Stokes equations. The computer code, called Transonic Navier-Stokes (TNS), uses a total of four zones for wing configurations (or can be extended to complete aircraft configurations by adding zones). In the inner blocks near the wing surface, the thin-layer Navier-Stokes equations are solved, while in the outer two blocks the Euler equations are solved. The diagonal algorithm yields a speedup of as much as a factor of 40 over the original algorithm/zonal method coded. The TNS code, in addition, has the capability to model wind tunnel walls. Transonic viscous solutions are obtained on a 150,000-point mesh for a NACA 0012 wing. A three-order-of-magnitude drop in the L2-norm of the residual requires approximately 500 iterations, which takes about 45 min of CPU time on a Cray-XMP processor. Simulations are also conducted for a different geometrical wing called WING C. All cases show good agreement with experimental data.

1. INTRODUCTION

Analytical methods and wind tunnel experiments are the two primary tools at the disposal of the design aerodynamicists today. Although yielding quick, closed-form solutions, analytical methods are limited to simple configurations. On the other hand, wind tunnel experiments can handle complicated configurations, but have flow limitations and both model fabrication and test time are expensive (Ref. 1). An alternative for the aerodynamicist, and one that eliminates some of the disadvantages associated with analytical methods and experimentation, is computational fluid dynamics (CFD).

Initial contributions to aircraft design made by CFD methods were due primarily to the use of linear panel methods or to the solution of various forms of the potential equation. This restriction was a result of existing computer storage and hardware limitations. Panel methods, however, were still useful in the design of aircraft, providing the flow was shock free. Solution of the potential equation extended the design of aircraft into the transonic regime, provided negligible amounts of entropy or vorticity were produced. Still, with these limitations, CFD tools coupled with optimizers have been instrumental in the partial design or improvement of some simple aircraft. Some examples are the Rockwell forward-swept wing (Ref. 1), the HiMAT wing (Ref. 2), the Airbus A-310 (Ref. 2), the Lockheed C-141B military transport (Ref. 2), and the Cessna Citation III business jet (Ref. 3).

To make full use of the three-dimensional Euler/Navier-Stokes formulations in aircraft design, many pertinent parameters must be accurately predicted (C_L , C_D , etc.). To achieve these predictions, high resolution is required to resolve the physics of shock/boundary-layer interaction, massive separation, and turbulent flow structures. However, even with the development of supercomputers such as the Cray-XMP and Cyber 205, there are still limitations due to storage-space and execution considerations. To overcome the computer storage problem, zonal approaches have become increasingly popular. By zonal approach we mean the partitioning of the flow field into distinct "zones," each of which is solved independently. The zonal approach has a number of advantages. First, a difficulty in generating three-dimensional grids for different types of complex configurations can be reduced with the use of zonal methods. Second, zonal methods would allow different types of grid topologies to be implemented where appropriate in order for the grids to be mesh efficient; (that is, more points on the configuration) where accuracy is desired, and fewer points in the flow field. The zonal concept has been successfully applied to the full potential equations (Ref. 4,5) and to the Euler equations (Ref. 6,7). It is also possible with the zonal approach to solve different types of equation sets in the different zones (Ref. 8).

*Research Scientist.

**Chief, Applied Computational Fluids Branch.

This paper discusses the computational approach in which the fast-convergent, Pulliam-Chaussee (Ref. 9) diagonal algorithm is coupled with a zonal approach. The new approach permits relatively inexpensive fine grid solutions to be made of the Euler/Navier-Stokes equations, which is especially important for flows with shock/boundary-layer interaction. Validation of the code by comparing numerical solutions with existing experimental data will also be presented.

2. GOVERNING EQUATIONS

The equations solved in this study are the Reynolds-averaged Navier-Stokes equations written in strong conservation law form. These equations are simplified by using the standard thin-layer approximation for the viscous terms. The thin-layer Navier-Stokes equations written in generalized curvilinear coordinates are

$$\partial_\tau \hat{Q} + \partial_\xi \hat{E} + \partial_\eta \hat{F} + \partial_\zeta \hat{G} = Re^{-1} \partial_\zeta \hat{S} \quad (1)$$

where

$$\hat{Q} = J^{-1} \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ e \end{bmatrix}, \quad \hat{E} = J^{-1} \begin{bmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho v U + \xi_y p \\ \rho w U + \xi_z p \\ U(e + p) - \xi_t p \end{bmatrix}$$

$$\hat{F} = J^{-1} \begin{bmatrix} \rho V \\ \rho u V + \eta_x p \\ \rho v V + \eta_y p \\ \rho w V + \eta_z p \\ V(e + p) - \eta_t p \end{bmatrix}, \quad \hat{G} = J^{-1} \begin{bmatrix} \rho W \\ \rho u W + \zeta_x p \\ \rho v W + \zeta_y p \\ \rho w W + \zeta_z p \\ W(e + p) - \zeta_t p \end{bmatrix}$$

with

$$U = \xi_t + \xi_x u + \xi_y v + \xi_z w$$

$$V = \eta_t + \eta_x u + \eta_y v + \eta_z w$$

$$W = \zeta_t + \zeta_x u + \zeta_y v + \zeta_z w$$

and

$$\hat{S} = J^{-1} \begin{bmatrix} 0 \\ \mu m_1 u_\zeta + (\mu/3) m_2 \zeta_x \\ \mu m_1 v_\zeta + (\mu/3) m_2 \zeta_y \\ \mu m_1 w_\zeta + (\mu/3) m_2 \zeta_z \\ \mu m_1 m_3 + (\mu/3) m_2 (\zeta_x u + \zeta_y v + \zeta_z w) \end{bmatrix}$$

Here $m_1 = \zeta_x^2 + \zeta_y^2 + \zeta_z^2$, $m_2 = \zeta_x u_\zeta + \zeta_y v_\zeta + \zeta_z w_\zeta$, and $m_3 = [(u^2 + v^2 + w^2)/2 + Pr^{-1}(\gamma - 1)^{-1}(a^2)]_\zeta$.

Pressure is related to the conservative flow variables Q by the equation of state

$$p = (\gamma - 1) \left[e - \frac{1}{2} \rho (u^2 + v^2 + w^2) \right]$$

The Beam-Warming algorithm (Ref. 10) is used to solve the governing equations; it is given by

$$\begin{aligned} & \left(I + h \delta_\xi \hat{A}^n - h D_i |_\xi \right) \left(I + h \delta_\eta \hat{B}^n - h D_i |_\eta \right) \\ & \left(I + h \delta_\zeta \hat{C}^n - h Re^{-1} \delta_\zeta J^{-1} \hat{M}^n J - h D_i |_\zeta \right) \Delta \hat{Q}^n = \\ & -h \left(\delta_\xi \hat{E}^n + \delta_\eta \hat{F}^n + \delta_\zeta \hat{G}^n - Re^{-1} \delta_\zeta \hat{S}^n + D_e \hat{Q}^n \right) \\ & = \hat{R}^n \end{aligned} \quad (2)$$

where A , B , C , and M are the Jacobian matrices $\partial E / \partial Q$, $\partial F / \partial Q$, $\partial G / \partial Q$, and $\partial S / \partial Q$, respectively. Note that M , which is derived from S , contains derivatives in η . For $h = \frac{1}{2}$ or 1, the integration is trapezoidal rule (second-order) or Euler implicit (first-order) in time. These equations are central-space-differenced and implicitly advanced in time. To maintain stability of the algorithm (because of the central-difference scheme used), an explicit fourth-order artificial dissipation term is added to the flux calculations and an implicit second-order dissipation term is added to each of the block tridiagonals. (The first working code (TNSCY3) used the above described algorithm incorporated in a zonal approach.)

For steady-state computations or first-order time integrations, a diagonal form of Eq. (2) is used. In this case, the left and right eigenvector matrices of \hat{A} , \hat{B} , and \hat{C} are used to diagonalize the one-dimensional operators. The diagonal algorithm in three dimensions has the form

$$T_\xi (I + h \delta_\xi \Lambda_\xi) \hat{N} (I + h \delta_\eta \Lambda_\eta) \hat{P} (I + h \delta_\zeta \Lambda_\zeta) T_\zeta^{-1} \Delta \hat{Q}^n = \hat{R}^n \quad (3)$$

For a complete derivation of the diagonal algorithm (as well definitions for \hat{N} , \hat{P} , etc.) see Ref. 9.

The main advantage of this form is the simplification of the matrix inversions from block tridiagonal inversions to scalar tridiagonal inversions. This simplification reduces the computational work by about 30% (Ref. 9). Also, the new scalar form for the inversion process allows the use of scalar pentadiagonal solvers so that the added fourth-order explicit artificial dissipation can be properly linearized and be made fully implicit. This form enhances stability and convergence rates (Refs. 11,12). To further enhance the convergence rate, a space-varying Δt has been used. It is given by the formula $\Delta t = \frac{\Delta t_0}{1+\sqrt{J}}$, Δt_0 , as used here, is simply a constant used to decrease or increase Δt . For most cases, the default value of Δt_0 was 5. (The modification of TNSCY3 by Eq. 3 and the spatially varying time steps constituted the new code, TNSCY4.) In viscous calculations the diagonal algorithm uses an explicit treatment of the viscous terms. The turbulence model used for all cases is the Baldwin-Lomax algebraic model (Ref. 13).

3. ZONAL APPROACH

To initiate the zonal approach, a coarse block is first generated about the configuration. For this case (Fig. 1) the geometry is a NACA 0012 wing with a sweep of 20°, aspect ratio of 3.0, taper ratio of 1.0, and dihedral and twist of zero. A total of four zones are used, zone 1 being the coarse (or base) block. Note also that the terms "zone" and "block" are used interchangeably. The coarse block can be generated either iteratively (Ref. 14), or through a marching scheme (Ref. 15). Both procedures have the capabilities of spacing and controlling orthogonality at the inner and outer boundaries, as well as simulating wind tunnel walls. The topology of the grid is H-type in both the spanwise and chordwise directions.

To generate the finer zone near the wing, a small zone of points about the wing is removed from zone 1. The space left open by the removal of points from zone 1 is then occupied by the finer grid (zone 2). Zone 2 is generated by putting twice as many points in every spatial direction relative to zone 1. This task is accomplished by cubic-spline interpolation of the coarse-grid points to the fine-grid points. To generate the viscous grids, a small zone of points is again removed about the wing from zone 2. Zones 3 and 4 now occupy the area left vacant by the removal of points from zone 2. Zone 3 occupies the area (vacated by zone 2) above the wing and includes the upper surface. Similarly, zone 4, occupies the area below the wing, including the lower surface of the wing. Zones 3 and zone 4 retain the same number of points in the streamwise and spanwise direction as does zone 2; however, points are further clustered in the normal direction to capture viscous effects. All zones overlap at the zonal boundaries, usually by one or two grid planes.

Figure 1 illustrates the wing grid and a chordwise slice of zones 1, 2 and 3 at the symmetry plane ($y = 0$). (The grid dimensions in each zone are also illustrated in Fig. 1.) Note the doubling of points in zone 2 (in the chordwise and normal directions) relative to zone 1. The doubling is also done in the spanwise directions, although this is not shown here. There can also be seen a one-to-one correspondence of points in the chordwise direction (also in the spanwise direction) between zones 2 and 3, and the clustering of grid cells in zone 3. Also highlighted are the wing surface, trailing-edge mesh, and wing tip region. Zone 4 is not shown, but it is essentially of the same structure as zone 3. For clarification, Fig. 2 shows a generic form of the overlapping procedure between two inviscid zones (this generic overlap procedure is the same between all zones, inviscid or viscous). Boundary conditions are established explicitly by means of this overlapping procedure. That is, boundary conditions for the zone 2, $\xi = A$ plane are obtained by interpolating data from the interior points of zone 1. And, conversely, boundary conditions for the zone 1, $\xi = B$ plane are obtained from the interior data from zone 2. The boundary conditions for a single surface are obtained with just a series of one-dimensional interpolations. The interpolation process is automated to the extent that only the two planes involved in the interpolation need to be defined, the base and target planes. Two interpolation schemes are coded; one is cubic spline and the other is linear. The cubic spline is best in smooth regions of the flow and the linear is best in the nonsmooth regions. The linear interpolation routine was used for the results presented herein. In Fig. 1, zones 1 and 2 are solved using the Euler equations (inviscid zones), and the Navier-Stokes equations are used in the viscous zones (zones 3 and 4).

The total number of grid points used is 149,071. The individual grid-point breakdown for each grid zone is as follows: grid 1, $65 \times 20 \times 19 = 24,700$; grid 2, $69 \times 29 \times 21 = 42,021$; grid 3 $61 \times 27 \times 25 = 41,175$; and grid 4, $61 \times 27 \times 25 = 41,175$. This grid has been constructed to fit the geometry used in the experiment of Lockman and Seegmiller (Ref. 16). Another grid with exactly the same characteristics for zones 2, 3, and 4, but with free-air boundaries specified for grid zone 1 has also been generated. The free-air grid is constructed from the wind tunnel wall grid by simply adding several grid surfaces to both the top and bottom of the wind tunnel wall grid. Therefore, the wind tunnel wall grid is an exact subset of the free-air grid. The number of points in the free-air grid (166,621) is slightly larger than the number in the wind tunnel wall grid. In a later section, flow fields computed on these two grids will be presented and compared with experiment. For more details and illustrations concerning the zonal grids, see Holst et al. (Ref. 17) and Flores (Ref. 18).

4. DATA MANAGEMENT

Once the grid is generated and divided into the proper zones, the flow solver is initiated. The iteration procedure starts in the outer Euler block (zone 1), proceeds to the inner Euler block (zone 2), and ends with the two Navier-Stokes blocks, first the upper Navier-Stokes block (zone 3) and then the lower block (zone 4). Only one iteration using a spatially varying time step is completed in each grid zone before passing to the next. However, many iteration and/or time-stepping strategies could be used to improve convergence.

Only the flow field solution (Q arrays), transformation Jacobian (J), metric quantities, and the turbulence model arrays (when appropriate) associated with a single block reside in the main memory of the Cray-XMP at one time. The information associated with the other blocks reside in extended storage. On the Cray XMP this device is called the Solid State Device (SSD). The SSD is used in the same way as rotating disk storage. However, the SSD storage is physically

made up of semiconductor memory and therefore is much faster. Using the SSD instead of disks greatly reduces I/O wait time, and for jobs which are normally I/O-bound this is a significant advantage.

In the present zonal approach, the use of the SSD allows a great deal of flexibility, since a larger number of grid blocks can be supported without significant additions to main memory. The limiting factor with regard to this point is the size of the SSD. The SSD installed with the NASA Ames Research Center Cray XMP has 16 million 64-bit words of memory. (The memory has recently been upgraded to 128 million 64-bit words.) This memory can easily be extended to 32 million words if half of the precision (32 bits) is used. The current version of the TNS code with grid dimensions as outlined earlier in the grid-generation section requires 5.8 million words of SSD storage. All arrays on the SSD are stored in 64-bit precision with the exception of the metric arrays, which are stored in 32-bit precision. A test was performed with 64-bit metric storage and produced results close to those using 32-bit storage.

To allow more space in the main memory, the metrics are shuffled into main memory from the SSD in two-dimensional planes as needed. This shuffling allows the maximum grid size of each grid zone to be about 50,000 points. Because the flow-solver algorithm used in TNS is an ADI-type algorithm with implicit sweeps in all three directions, the metrics must be shuffled into main memory with three different orientations, in x-y planes, x-z planes, and y-z planes. Thus, there are three different copies of the metrics stored on the SSD corresponding to the three different metric orientations. Because of the availability of so much storage on the SSD this duplication causes no problems and makes the overall memory management more efficient.

In addition, each of the metric arrays is required in main memory several times for each grid zone during each iteration. This constant demand of arrays places extreme demands on the TNS I/O requirements. Nevertheless, with the efficiency level afforded by the SSD, these I/O requirements are handled with no problems. For example, a typical solution computed with the TNS code involving about 150,000 grid points requires about 500 iterations to reduce the L2 norm of the residuals in each grid zone by about three orders of magnitude. During this typical run, 5.97 billion 64-bit-equivalent words are transferred between the Cray XMP and the SSD. This transfer requires about 2780 sec of cpu time and only about 40 sec of SSD I/O time. In contrast, it is estimated that the same calculation using disk instead of SSD would require about 10 hr of I/O time. (See ref. 19, for tables of tabulated I/O times of disk versus SSD.)

5. DISCUSSION OF RESULTS

The first case tested consisted of the NACA 0012 wing, previously described, with the following flow conditions; $M_\infty = 0.826$, $\alpha = 2.0^\circ$, and a Reynolds number based on a chord of 8 million. This case is moderately difficult, involving a strong shock which extends from zone 3 (viscous upper-surface zone) into zones 2 and 1 (inviscid zones). With the zonal approach, the test case was run on a 150,000-point mesh, which for three dimensions is a relatively fine grid.

For this case the wind tunnel wall effects are very significant. This effect can be seen in Fig. 3 where the pressure coefficient distributions from TNS, with and without the walls modeled, are compared with experiment (Ref. 16). The shift in shock position caused by the tunnel walls is obvious. The shock position for the case with walls is in good agreement with the experimental shock position whereas the free-air shock position is too far upstream by about 10-20% of chord.

General agreement between the wind tunnel wall case and experiment is better inboard of mid-semispan than it is outboard. In particular, the computed upper surface shock strength at $2y/b = 0.79$ is larger than that of the experiment. This larger strength is caused by a large boundary-layer separation in the experimental results at this semispan location, which is not accurately reproduced by the computed results. A good picture of this situation is given in Figs. 4 and 5 which show a set of computed particle paths (Fig. 4) and an oil-flow photograph taken from the experiment (Fig. 5). The experimental separation is about twice as large as the computed separation. The spanwise extent of the experimental separation is reasonably predicted by the computation, but the streamwise extent is underpredicted. Some reasons for this discrepancy are coarse-grid and turbulent-model effects, as well as the sensitivity of the flow pattern to changes in the free-stream Mach number. Despite the difference in the size of the separation zone, the overall comparison is quite encouraging.

The convergence rate of TNSCY4 versus TNSCY3 is illustrated in Fig. 6. The time step used in the nondiagonal version was $\Delta t = 0.004$, which was the largest time step possible while still maintaining stability of the code. The diagonal version (TNSCY4) used a variable time step (as described previously). (Even though we are comparing a fixed time step solution and a variable time step solution, the main speedup in the diagonal algorithm is not in the variable time stepping procedure, but in the proper linearization of the dissipation terms (Ref. 11). The slow rate of convergence in the nondiagonal version (TNSCY3) seems to occur in the outer inviscid zones. The residual in the viscous zones in the first thousand iterations drops fairly fast, then begins to flatten out. In 5000 iterations, all zones have dropped about two orders of magnitude in the L2 norm of the residual. In contrast, the convergence rate of the diagonal version drops rapidly in all the zones. A three-order-of-magnitude drop in the L2 norm occurs in about 400-500 iterations. This convergence rate (coupled with the decrease in arithmetic operation count caused by the diagonal algorithm) increases by a factor of 40 the speed with which solutions are obtained. The faster convergence rate, as stated before, is due to the proper linearization of the fourth-order-explicit dissipation operator. This rate was possible in the nondiagonal version, but would involve inverting block pentadiagonals, which would substantially increase the computational cost. More time was required for the same case when wind tunnel walls were modeled, since a stronger shock occurs.

We can also look at the development of lift and number of supersonic points (NSP) to study the convergence characteristics. Figure 7 shows the development stages, the X-axis is the number of iterations, while the Y-axis is left unlabeled (since the actual values were not of significance, but the overall trend in the approach to the steady state is important). The NSP overshoots the final converged solutions at about 100 iterations, but then quickly approaches the converged solution. At about 200 iterations, it is within 7% of the converged solution and at 300 iterations it is within 1%. The lift also overshoots the final converged value, but at 200 iterations it is within about 4% of the final solution and at 300 iterations it is within 1%. If, instead of a three-order-of-magnitude drop in residual, convergence is based on 95% of the converged lift, then a solution can be generated in about 200 iterations, or 18 min of CPU time for this fine-grid calculation.

The next case presented consists of a massive, shock-induced, boundary-layer separation. This case was computed to ascertain the degree of robustness of the present algorithm and, in particular, the ability of the present zonal interface scheme to cope with large flow gradients. The geometry used is the same as that of the last case. The free-stream Mach number and angle of attack have been arbitrarily chosen to be 0.9 and 5° , respectively. Utilization of the wind tunnel wall boundaries produced a "choked" solution with a shock wave spanning the tunnel. After several hundred iterations and a moderately converged calculation, the solution diverged as expected. This result was a consequence of the "fixed" upstream boundary conditions forcing more mass flow through the tunnel than the choked condition would allow.

The solution was repeated with free-air boundary conditions, and then convergence was easily achieved. Computed particle paths on the upper wing surface are displayed in fig. 8. Note that approximately half of the upper wing surface is separated. The solution contains several interesting features, including a separation saddle point, a focus, a reattachment saddle point, and a node. Note also that this computed solution has a stable sequence of critical points on the separation line; that is, a node followed by a saddle point and then a focus. This case required about 2.8 hr of CPU time.

Two different perspective views of the three-dimensional particle paths are shown in fig. 9. Figure 9a shows a view from outboard of the wingtip and fig. 9b shows a view from behind and above the wing. The height and three-dimensionality of the separation zone are apparent in these figures. The dashed particle paths move along the wing surface until the separation line is encountered and then are deflected up over the separation bubble, with a few of the dashed paths captured by the primary swirling flow at the center of the wing. The solid particle paths are more intimately involved with the two swirling pockets of flow and essentially define these regions.

The position of the separation region relative to the zonal interface boundary is best displayed by plotting particle paths constrained to lie in spanwise cross-sectional planes. Two such plots are displayed in fig. 10. Figure 10a shows cross-sectional particle paths for a semispan station of $2y/b = 0.66$ ($k=13$). At $k=13$ the separation region is large and easily extends above the zonal boundary from the Navier-Stokes region into the Euler region. Nevertheless, the solution looks qualitatively reasonable. An enlargement of the separated portion of the solution is shown in fig. 10b. From this figure it can be seen that the particle paths pass smoothly across the interface boundary with no function or slope discontinuities. Thus, the primary objective of doing this calculation was achieved. Despite the existence of a strong shear gradient across the explicitly updated interface boundary, the present approach is capable of predicting a stable solution that is reasonably free from interface boundary influence.

The next computation involves the WING C configuration. WING C is a generic advanced-technology wing designed by Ames Research Center and the Lockheed Georgia Company. Two sets of experiments are available for this configuration, one by Hinson and Burdges (Ref. 20) and one by Keener (Ref. 21). WING C, as illustrated in figure 11, it has an aspect ratio of 2.6, a twist angle of 8.17° , a taper ratio of 0.3, and a leading-edge sweep of 45° .

The separated-flow case consists of the WING C design conditions: $M_\infty = 0.85$, $\alpha = 5^\circ$, and $Re = 6.8 \times 10^6$. These conditions were intended to result in attached flow with a mild shock wave and a mild pressure recovery. But, for reasons discussed in Ref. 21, these conditions produced a "local" (so defined by its author) flow separation.

The calculated pressure coefficient distributions are compared with experimental data in Fig. 12. Note that, in this figure, the angle of attack for the Keener data (Ref. 21) is 5° , whereas for the Hinson and Burdges data (Ref. 20), it is 5.9° in accordance with their accounting for wall effects. Consequently, computations were done for these two angles of attack ($\alpha = 5^\circ$ and 5.9°) at $M_\infty = 0.85$, and $Re = 6.8 \times 10^6$. The computation with $\alpha = 5^\circ$ agrees with the two experiments better in terms of matching leading-edge pressures. On the other hand, the results with $\alpha = 5.9^\circ$ seem to agree well with experiments in terms of shock position and lower-surface pressures. Note that the differences between the two computations and the two experiments are generally the same order of magnitude. The experimental differences again may be explained by considering the uncertainties in the measurements of Mach number and angle of attack, and the assessment of the wall-interference effects associated with the experiments. Possible distortions in the wing geometries caused by aerodynamic forces and moments during the experiments could also have affected the measurements. On the other hand, the numerical simulation has its own difficulties associated with the coarse grid near the tip, an inadequate turbulence model, free-air boundary conditions, etc. However, the overall comparison is good considering the uncertainties just mentioned.

The experimental oil-flow picture for the same case is presented in Fig. 13. The computed skin-friction lines for this case are presented in Fig. 14. Note that the larger angle of attack ($\alpha = 5.9^\circ$) was used in the computation to get closer computational/experimental agreement. The calculation with $\alpha = 5^\circ$ did not display a separated flow. In this computation, the global features of the experiment are predicted well. The location and size of the separation line, the streamlines being trapped by the vortex-like formation inboard of the separation line, the curvature of the tip streamlines, and almost two-dimensional flow outside the separation zone are all accurately predicted. The critical points of the skin-friction map were not well reproduced, but a nodal point (N_s) along the line of separation and near the tip is predicted. Overall, the simulation is globally good and is encouraging.

6. CONCLUSIONS

A fast diagonal algorithm has been successfully implemented within the framework of a zonal approach. Results indicate that the modified code (in obtaining a solution for a moderately difficult case) still maintains its fast convergence characteristics. This improvement is demonstrated by producing a three-dimensional fine-grid Euler/Navier-Stokes solution in 45 min on the Cray-XMP, which is a speedup by a factor of 40 over the original code. The effects of the overlapping procedure as well as the explicit boundary conditions used in the zonal approach do not appear to degrade the convergence characteristics of the diagonal scheme. Numerical results were compared to experimental results for moderate to difficult flow conditions for two different low-aspect wings. Good agreement existed between the associated pressure fields. Good correlation between the computed separated surface flow and the experimental oil flow patterns was also obtained.

7. REFERENCES

1. Kutler, P., "A Perspective of Theoretical and Applied Computational Fluid Dynamics," AIAA Paper 83-0037, Jan., 1983.
2. Ballhaus, W. F., "Supercomputing in Aerodynamics," presented at the Conference on Frontiers of Supercomputing, 1983.
3. Cosentino, G. B. and Holst, T. L., "Numerical Optimization Design of Advanced Transonic Wing Configurations," AIAA Paper 85-0424, Jan., 1985.
4. Atta, E. H., "Component-Adaptive Grid Interfacing," AIAA Paper 81-0382, Orlando, Fla., 1982.
5. Atta, E. H. and Vadyak, J. A., "Grid Interfacing Zonal Algorithm for Three-Dimensional Transonic Flows about Aircraft Configurations," AIAA Paper 82-1017, 1982.
6. Hessenius, K. A. and Pulliam, T. H., "A Zonal Approach to Solution of the Euler Equations," AIAA Paper 82-0969, July, 1984.
7. Rai, M. M., "A Conservative Treatment of Zonal Boundaries for Euler Calculations," AIAA Paper 84-0164, Jan., 1984.
8. Chaderjian, N. M. and Steger, J. L., "A Zonal Approach for the Steady Transonic Simulation of Inviscid Rotational Flow," AIAA Paper 83-1927, July, 1983.
9. Pulliam, T. H. and Chaussee, D. S., "A Diagonal Form of An Implicit Approximate-Factorization Algorithm," *J. Comp. Phys.*, Vol. 39, No. 2, 1981, pp. 347-363.
10. Beam, R. M. and Warming, R. F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation Law Form," *J. Comp. Phys.*, Vol. 22, 1976, pp. 87-110.
11. Pulliam, T. H., "Artificial Dissipation Models for the Euler Equations," AIAA Paper 85-0438, Jan., 1985.
12. Pulliam, T. H. and Steger, J. L., "Recent Improvements in Efficiency, Accuracy, and Convergence for Implicit Approximate-Factorization Algorithms," AIAA Paper 85-0360, Reno, Nev., 1985.
13. Baldwin, B. S. and Lomax, H., "Thin-Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78-257, Jan. 1978.
14. Sorenson, R. L., "Three-Dimensional Elliptic Grid Generation About Fighter Aircraft For Finite-Difference Solutions," to be published as a NASA Technical Manual.
15. Edwards, T. A., "Noniterative Three-Dimensional Grid Generation Using Parabolic Partial Differential Equations," AIAA Paper 85-0485, Jan., 1985.
16. Lockman, W. K. and Seegmiller, H. L., "An Experimental Investigation of the Subcritical and Supercritical Flow About a Swept Semispan Wing," NASA TM-84367, June 1983.
17. Holst, T. L., Kaynak, U., Gundy, K. L., Thomas, S. D., Flores, J., and Chaderjian, N. M., "Numerical Solution of Transonic Wing Flows Using An Euler/Navier-Stokes Zonal Approach," AIAA Paper 85-1640, July, 1985.
18. Flores, J., "Convergence Acceleration for a Three-Dimensional Euler/Navier-Stokes Zonal Approach," AIAA Paper 85-1495, July, 1985.
19. Holst, T. L., Thomas, S. D., Kaynak, U., Gundy, K. L., and Flores, J., "Computational Aspects of Zonal Algorithms for Solving the Compressible Navier-Stokes Equations in Three Dimensions," NASA TM 86774, Oct., 1985.
20. Hinson, B. L. and Burdges, K. P., "Acquisition and Application of Transonic Wing and Far-Field Test Data for Three-Dimensional Computational Method Evaluation, Vol. II. Appendix B, Experimental Data," Lockheed Georgia Company, AFOSR-TR-80-0422, 1980.
21. Keener, E. R., "Computational-Experimental Pressure Distributions on a Transonic, Low-Aspect-Ratio Wing," AIAA Paper 84-2092, Aug. 1984.

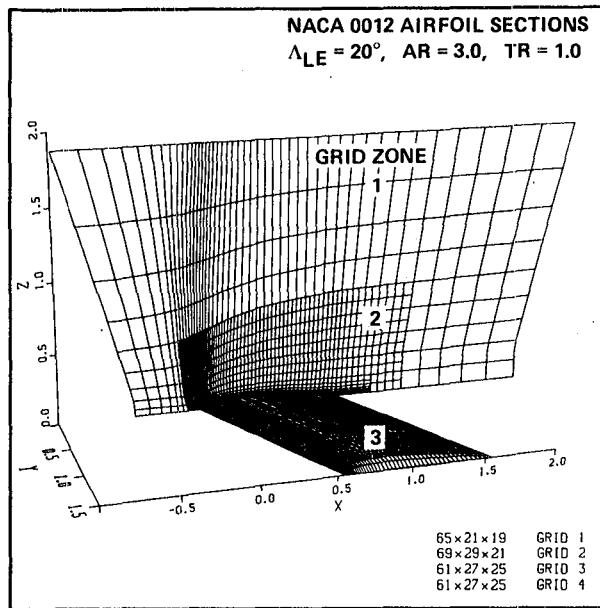


Fig. 1 Perspective view of embedded grids.

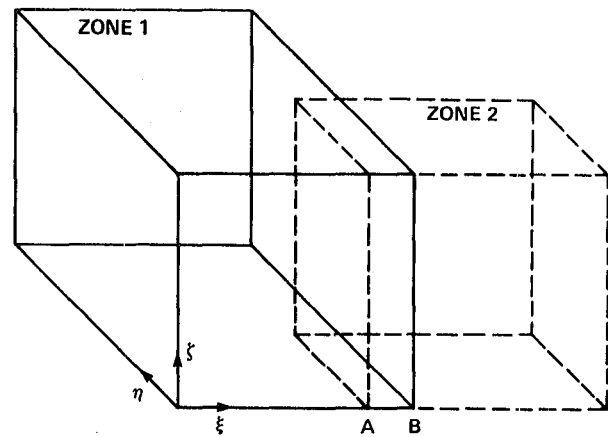


Fig. 2 Two-zone grid arrangement showing overlap.

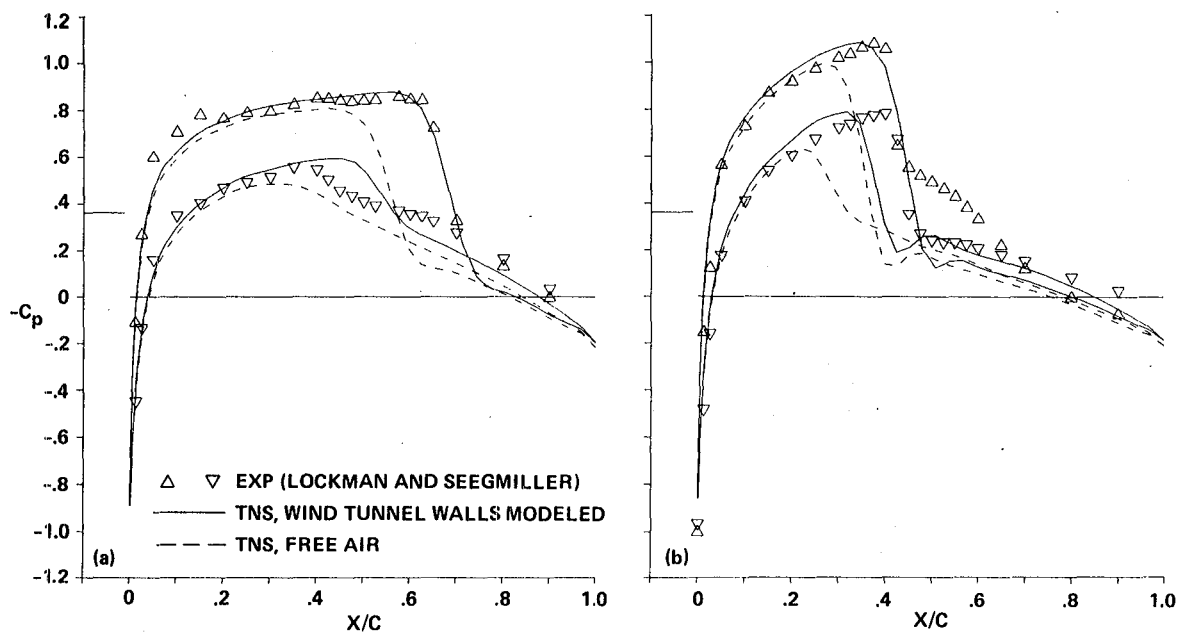


Fig. 3 Pressure coefficient comparisons: NACA 0012 airfoil section, $\Lambda_{LE} = 20^\circ$, $AR = 3.0$, $TR = 1.0$, $M_\infty = 0.826$, $\alpha = 2.0^\circ$, $Re = 8 \times 10^6$. a) $2y/b = 0.25$; b) $2y/b = 0.78$.

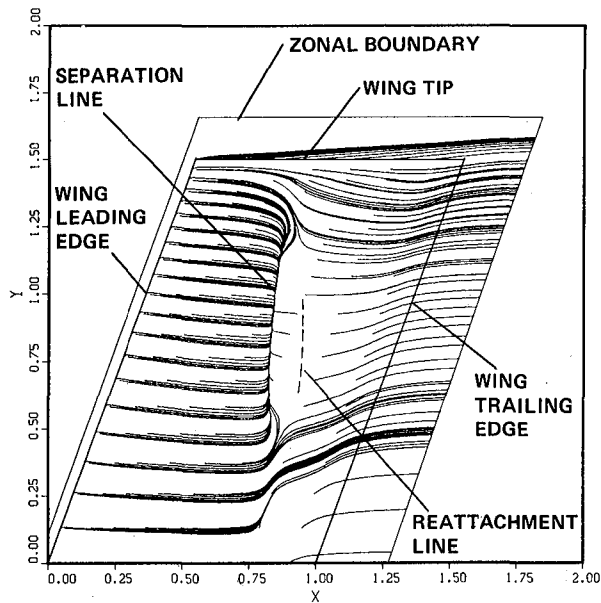


Fig. 4 Computed particle paths on the upper wing surface: NACA 0012 airfoil section, $\Lambda_{LE} = 20^\circ$, $AR = 3.0$, $TR = 1.0$, $M_\infty = 0.826$, $\alpha = 2.0^\circ$, $Re = 8 \times 10^6$.

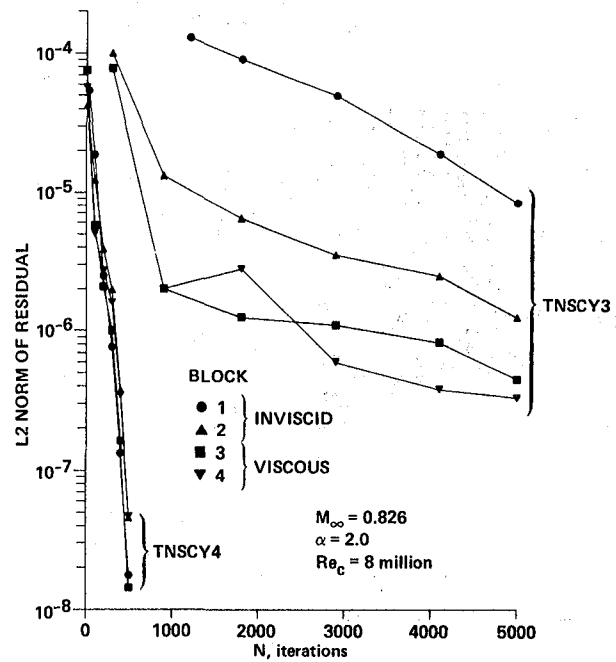


Fig. 6 Convergence rate comparison

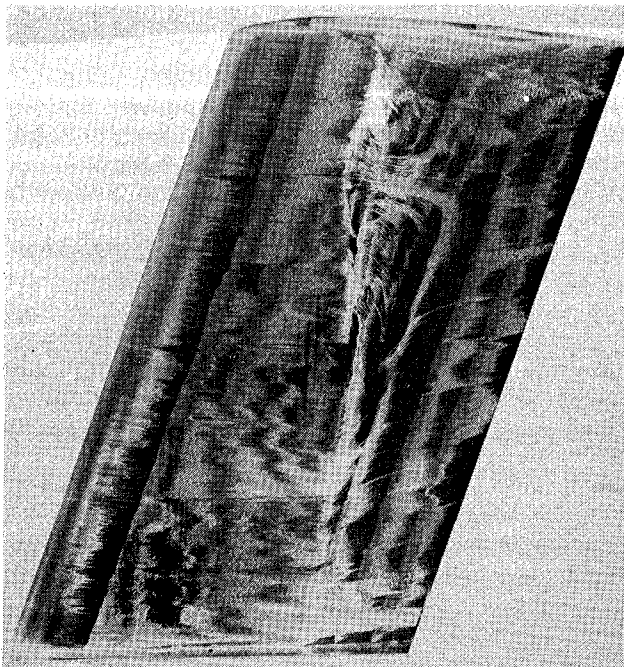


Fig. 5 Oil-flow pattern on upper wing surface: NACA 0012 airfoil section, $\Lambda_{LE} = 20^\circ$, $AR = 3.0$, $TR = 1.0$, $M_\infty = 0.826$, $\alpha = 2.0^\circ$, $Re = 8 \times 10^6$. (From Ref. 16)

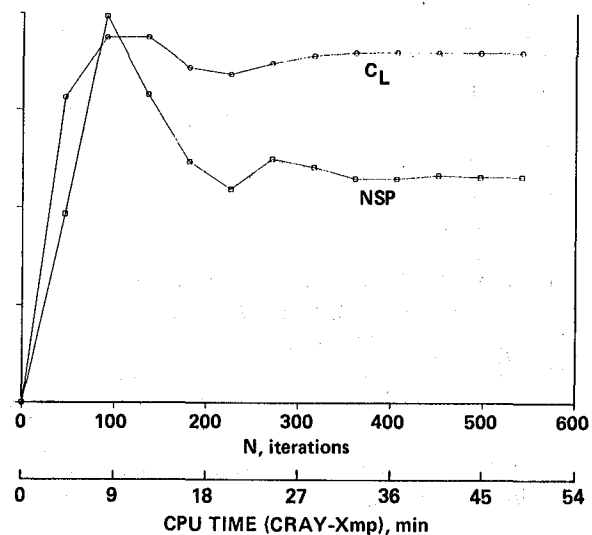


Fig. 7 Development of NSP and C_L .

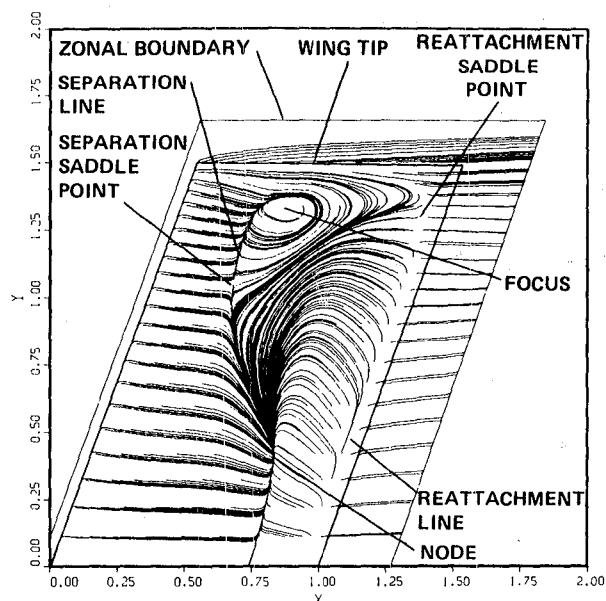


Fig. 8 Computed particle paths on the upper wing surface: NACA 0012 airfoil section, $\Lambda_{LE} = 20^\circ$, $AR = 3.0$, $TR = 1.0$, $M_\infty = 0.9$, $\alpha = 5.0^\circ$, $Re = 8 \times 10^6$.

PARTICLES RELEASED JUST ABOVE SURFACE

----- $X/C = 0.05$
 ——— $X/C = 0.40$

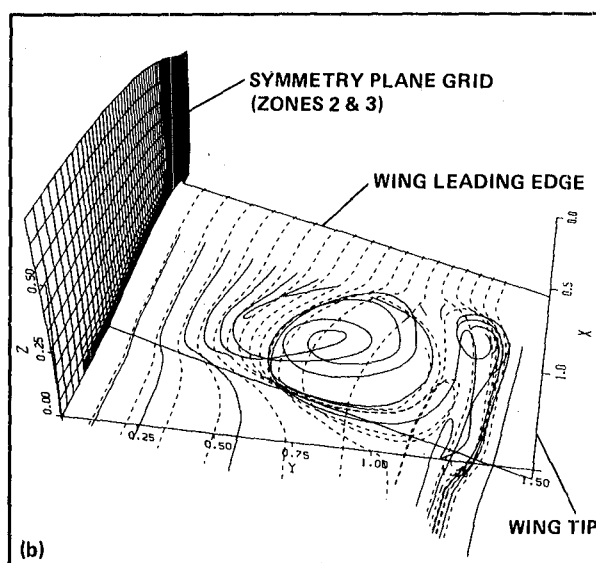
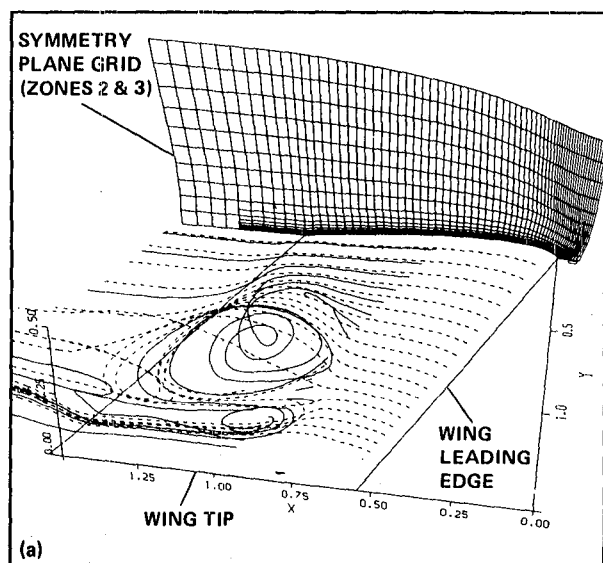


Fig. 9 Computed three-dimensional particle paths on the upper wing surface: NACA 0012 airfoil section, $\Lambda_{LE} = 20^\circ$, $AR = 3.0$, $TR = 1.0$, $M_\infty = 0.9$, $\alpha = 5.0^\circ$, $Re = 8 \times 10^6$. a) View from outboard of wingtip; b) View from behind and above wing.

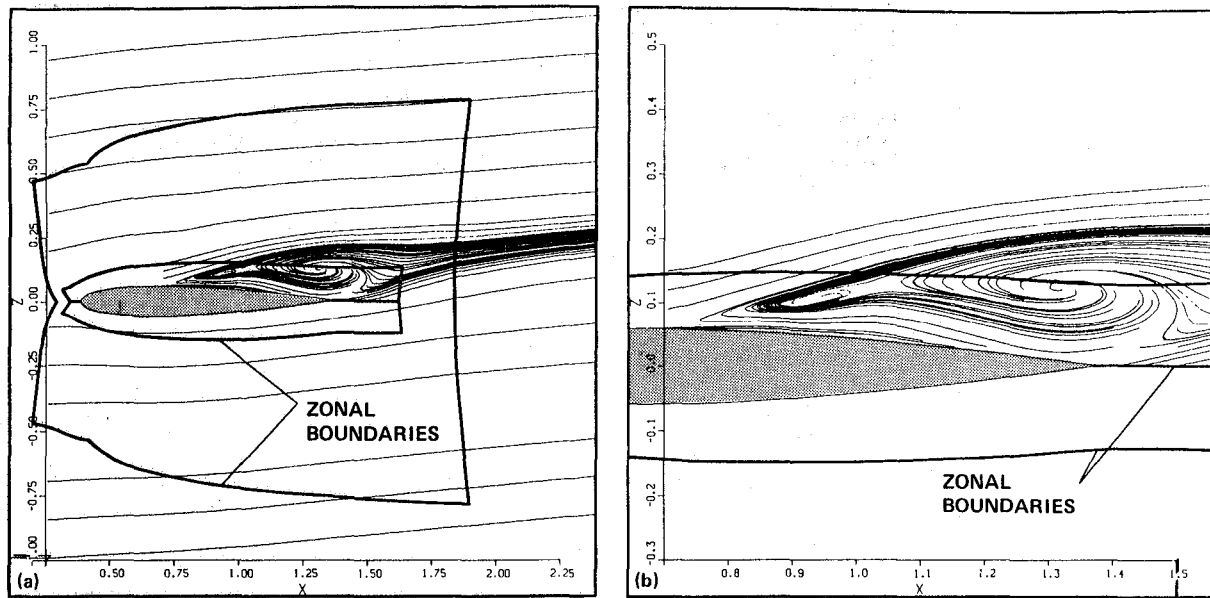
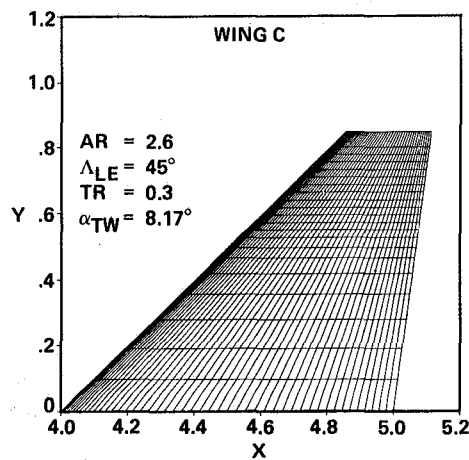


Fig. 10 Computed cross-sectional particle paths: NACA 0012 airfoil section, $\Lambda_{LE} = 20^\circ$, $AR = 3.0$, $TR = 1.0$, $M_\infty = 0.9$, $\alpha = 5.0^\circ$, $Re = 8 \times 10^6$. a) $2y/b = 0.66$; b) $2y/b = 0.66$, Enlarged view of separated flow region.



EXPERIMENTAL DATA:

- (a) KEENER
NASA-Ames
(LARGE SCALE MODEL)
- (b) HINSON & BURDGES
LOCKHEED-GEORGIA
(SMALL SCALE MODEL)

Fig. 11 Planform view of WING C: $\Lambda_{LE} = 45^\circ$, $AR = 2.6$, $TR = 0.3$, $\alpha_{TWIST} = 8.18^\circ$.

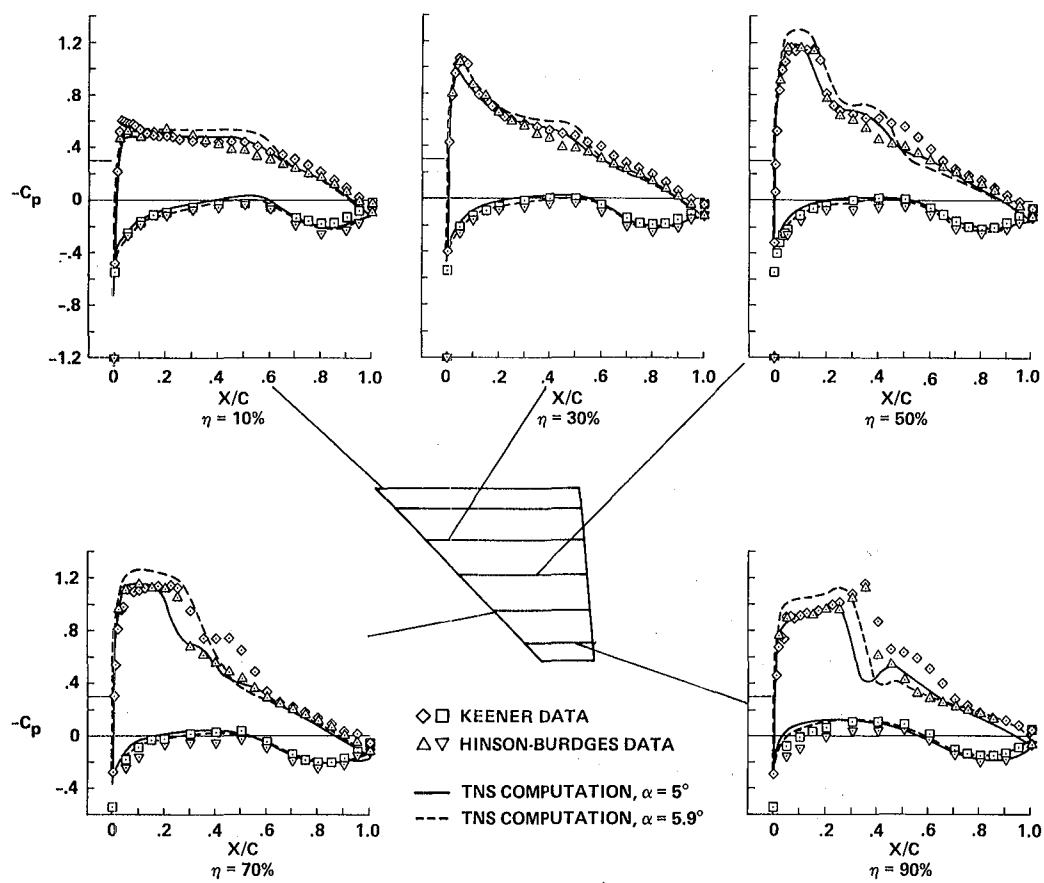


Fig. 12 Comparison of experimental and computed pressure coefficients for WING C: $M_\infty = 0.85$, $Re_{m.a.c.} = 6.8 \times 10^6$.

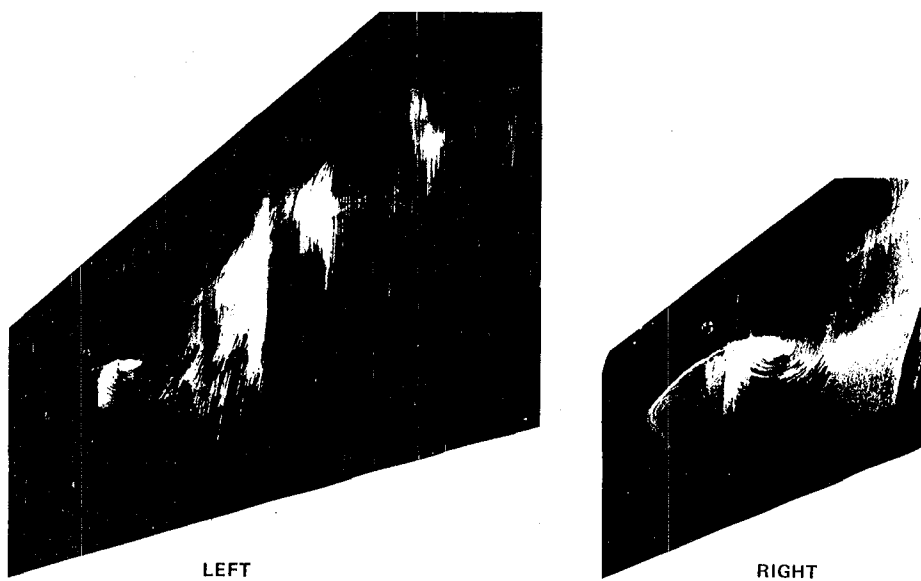


Fig. 13 Experimental oil-flow picture (from Ref. 21) and an expanded view of the counter-rotating vortices for WING C: $M_\infty = 0.85$, $\alpha = 5^\circ$, $Re_{m.a.c.} = 6.8 \times 10^6$.

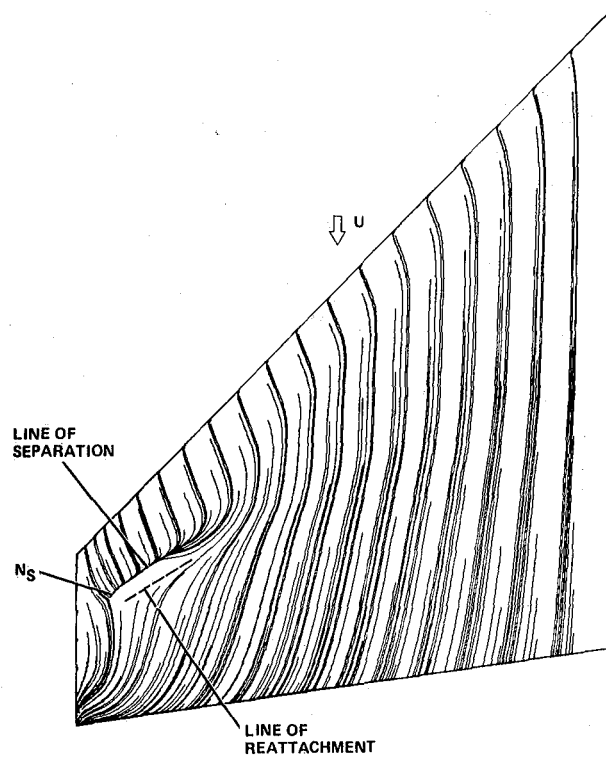


Fig. 14 Computed skin-friction lines for WING C upper surface: $M_\infty = 0.85$, $\alpha = 5.9^\circ$, $Re_{m.a.c.} = 6.8 \times 10^6$.

1. Report No. NASA TM-88248		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle TRANSONIC NAVIER-STOKES WING SOLUTION USING A ZONAL APPROACH: PART 1. SOLUTION METHODOLOGY AND CODE VALIDATION				5. Report Date March 1986	
				6. Performing Organization Code	
7. Author(s) J. Flores, T. L. Holst, Unver Kaynak (Sterling Software Palo Alto), K. Gundy, and S. D. Thomas (Sterling Software, Palo Alto, CA)				8. Performing Organization Report No. A-86206	
9. Performing Organization Name and Address Ames Research Center Moffett Field, CA 94035				10. Work Unit No.	
				11. Contract or Grant No.	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546				13. Type of Report and Period Covered Technical Memorandum	
				14. Sponsoring Agency Code 505-60	
15. Supplementary Notes Point of Contact: Jolen Flores, Ames Research Center, MS 202A-14, Moffett Field, CA 94035 (415) 694-5369 or FTS 464-5369					
16. Abstract A fast diagonalized Beam-Warming algorithm is coupled with a zonal approach to solve the three-dimensional Euler/Navier-Stokes equations. The computer code, called Transonic Navier-Stokes (TNS), uses a total of four zones for wing configurations (or can be extended to complete aircraft configurations by adding zones). In the inner blocks near the wing surface, the thin-layer Navier-Stokes equations are solved, while in the outer two blocks the Euler equations are solved. The diagonal algorithm yields a speedup of as much as a factor of 40 over the original algorithm/zonal method coded. The TNS code, in addition, has the capability to model wind tunnel walls. Transonic viscous solutions are obtained on a 150,000-point mesh for a NACA 0012 wing. A three-order-of-magnitude drop in the L2-norm of the residual requires approximately 500 iterations, which takes about 45 min of CPU time on a Cray-XMP processor. Simulations are also conducted for a different geometrical wing called WING C. All cases show good agreement with experimental data.					
17. Key Words (Suggested by Author(s)) Navier-Stokes Zonal Approach Transonic wing solutions				18. Distribution Statement Unlimited Subject Category - 02	
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 15	
				22. Price* A02	

End of Document